ABSTRACT OF THE DISCLOSURE

[1044]   Program code executed in an environment in which latency exists between an execution event and detection of the execution event may be profiled using a technique that includes backtracking from a point in a representation of the program code, which coincides with the detection toward a preceding operation associated with the execution event. Backtracking identifies the preceding operation at a displacement from the detection point unless an ambiguity creating location is disposed between the detection point and the preceding operation. In general, the relevant set of ambiguity creating locations is processor implementation dependent and program code specific; however, branch targets locations, entry points, and trap or interrupt handler locations are common examples. In some realizations, the techniques may be used to associate cache miss (or hit) information with execution of particular memory access instructions. However, more generally, such techniques may be employed to associate observed execution characteristics with particular instructions of program code or associated operations based on event detections that may, in general, lag execution of the triggering instruction or operation by an interval that allows intervening program flow ambiguity.